

Homework 1: Natural Deduction

15-814: Type Systems for Programming Languages

TA: Kumar Avijit (kavijit@cs.cmu.edu)

Out: September 15, 2009

Due: September 22, 2009 (midnight)

Welcome to 15-814! In this homework, you will be studying (and proving) properties of a system of natural deduction for logic. Theorems about deductive systems are often proven using rule induction. Thus by the end of this homework, we expect you to become comfortable with induction arguments, and also understand the basics of natural deduction.

The homework should be submitted electronically as a single PDF file named `solution.pdf`. Submit the homework by copying the file(s) to the directory

`/afs/cs.cmu.edu/academic/class/15814/handin/⟨user-id⟩/hw1`

1 Structural Properties

In this question, we shall prove structural properties about the natural deduction system studied in the class for Intuitionistic Minimal Logic **M**. Let us first recap the system. We use the following judgment forms:

1. $A \text{ true}$ denotes that the proposition A is true categorically.
2. $A_1 \text{ true}, \dots, A_n \text{ true} \vdash A \text{ true}$ iff A is derivable assuming that A_1 through A_n are derivable. We shall abbreviate the context using upper-case letters Γ, Δ, \dots . These denote *finite sets* of hypotheses.

Let us consider a fragment of logic with only implication (\supset).

$$\frac{}{\Gamma_1, A \text{ true}, \Gamma_2 \vdash A \text{ true}} (\text{hyp}) \qquad \frac{\Gamma, A \text{ true} \vdash B \text{ true}}{\Gamma \vdash A \supset B \text{ true}} (\supset\text{-I})$$

$$\frac{\Gamma \vdash A \supset B \text{ true} \quad \Gamma \vdash A \text{ true}}{\Gamma \vdash B \text{ true}} (\supset\text{-E})$$

We call a rule $\frac{\mathcal{J}_1 \dots \mathcal{J}_n}{\mathcal{J}}$ *admissible* under a rule set \mathcal{R} if \mathcal{J} is derivable from \mathcal{R} whenever \mathcal{J}_1 through \mathcal{J}_n are derivable from \mathcal{R} . Please refer to Section 2.2 of PFPL for a description of admissibility.

Task 1 (Weakening) Show that if a proposition is derivable under a context Γ , then it is derivable under any context Γ' such that $\Gamma' = \Gamma, A \text{ true}$ for any proposition A . That is, prove that the following rule is admissible:

$$\frac{\Gamma \vdash B \text{ true}}{\Gamma, A \text{ true} \vdash B \text{ true}} (\text{weaken})$$

You are free to use the result about exchange proved above.

Task 2 (Contraction) Now we shall show that duplicate assumptions do not affect derivability. Show that the following rule is admissible:

$$\frac{A \text{ true}, A \text{ true}, \Gamma \vdash B \text{ true}}{A \text{ true}, \Gamma \vdash B \text{ true}}$$

Task 3 (Transitivity) We will now show that the consequence relation is transitive. As discussed in the class, there are various equivalent ways to state this. In its simplest form, we want to prove the following rule to be admissible:

$$\frac{A \text{ true} \vdash B \text{ true} \quad B \text{ true} \vdash C \text{ true}}{A \text{ true} \vdash C \text{ true}} (\text{trans}_1)$$

The contextual version of this rule is the following:

$$\frac{\Gamma \vdash B \text{ true} \quad \Delta, B \text{ true} \vdash C \text{ true}}{\Gamma, \Delta \vdash C \text{ true}} (\text{trans}_2)$$

Observe that Rule (trans_1) can be derived from Rule (trans_2) by choosing $\Gamma = A \text{ true}$ and $\Delta = \cdot$.

In presence of weakening and contraction, it is sufficient to state the above rule with $\Gamma = \Delta$. This is the version we shall prove to be admissible.

Show that the following rule is admissible:

$$\frac{\Gamma \vdash A \text{ true} \quad \Gamma, A \text{ true} \vdash B \text{ true}}{\Gamma \vdash B \text{ true}} (\text{trans})$$

2 Proof terms

We now extend our system in two ways: first, we add rules for conjunction (\wedge) and disjunction (\vee) to our logic. Second, we also augment the truth judgment $A \text{ true}$ with proof terms. The new judgment is written as $m:A$. The context Γ now consists of hypotheses of the form $x:A$, where x is a variable. We recap the

natural deduction system for \supset, \wedge, \vee fragment of the logic with proof terms.

$$\begin{array}{c}
\frac{x\#\Gamma \quad x\#\Delta}{\Gamma, x:A, \Delta \vdash x:A} (\text{hyp}^+) \qquad \frac{\Gamma, x:A \vdash m:B \quad x\#\Gamma}{\Gamma \vdash \lambda x:A. m:A \supset B} (\supset\text{-I}^+) \\
\\
\frac{\Gamma \vdash m_1:A \supset B \quad \Gamma \vdash m_2:A}{\Gamma \vdash \text{apply}(m_1, m_2):B} (\supset\text{-E}^+) \qquad \frac{\Gamma \vdash m_1:A \quad \Gamma \vdash m_2:B}{\Gamma \vdash \langle m_1, m_2 \rangle:A \wedge B} (\wedge\text{-I}^+) \\
\\
\frac{\Gamma \vdash m:A \wedge B}{\Gamma \vdash \text{fst } m:A} (\wedge\text{-E}_1^+) \qquad \frac{\Gamma \vdash m:A \wedge B}{\Gamma \vdash \text{snd } m:B} (\wedge\text{-E}_2^+) \\
\\
\frac{\Gamma \vdash m:A}{\Gamma \vdash \text{inl}_{A,B} m:A \vee B} (\vee\text{-I}_1^+) \qquad \frac{\Gamma \vdash m:B}{\Gamma \vdash \text{inr}_{A,B} m:A \vee B} (\vee\text{-I}_2^+) \\
\\
\frac{\Gamma \vdash m:A \vee B \quad \Gamma, x:A \vdash m_1:C \quad \Gamma, y:B \vdash m_2:C \quad x\#\Gamma \quad y\#\Gamma}{\Gamma \vdash \text{case } m\{\text{inl } x \Rightarrow m_1 \mid \text{inr } y \Rightarrow m_2\}:C} (\vee\text{-E}^+)
\end{array}$$

The above rules mimic the proof-term-free rules from Section 1. There is a little bureaucratic overhead involved in maintaining variables. We follow a variable-hygiene convention that all variables in a context are unique. For example, in Rule $(\supset\text{-I}^+)$, we insist that x should not already appear in the context ($x\#\Gamma$). In case of clashes, we can always α -vary.

2.1 α -equivalence, free and bound variables

We have used variables in the context to denote occurrences of hypotheses. The actual names of the variables do not matter. This is akin to saying that the name of the variable x in the definition of a function $f : x:\mathbb{R} \mapsto x + 1$, does not matter. One could variously write this definition as $f : y:\mathbb{R} \mapsto y + 1$ replacing the bound variable x by y . This process is called α -conversion. For a formal definition of α -conversion, please refer to Chapter 6 of PFPL.

2.2 Substitution

A free variable $x:A$ in a proof stands for zero or more undischarged hypotheses of the proposition A . We now formalize the notion of a proof m_1 of A discharging an assumption $x:A$ in another proof m_2 . This process simply substitutes m_1 recursively wherever there is a free x in m_2 . The definition is by induction on the structure of m_2 . The tricky part in substitution is that the result of substitution is defined only upto α -equivalence. Consider the substitution $[x/y]\lambda x:A. \text{apply}(x, y)$. The variable x occurring inside λ -term is *bound* by the λ and is different from the *free* x outside. Thus the result of the substitution is not $\lambda x:A. \text{apply}(x, x)$, because doing so causes the free x to also become bound by the λ . To avoid this capture, we need to first rename the bound variable x to a fresh x' . Doing so, we obtain the result $\lambda x':A. \text{apply}(x', x)$. This operation of

renaming bound variables while doing a substitution in the body of the binder is called capture-avoiding substitution. For example, while substituting m_2 for x in $\lambda y:A.m_1$, we α -vary the binder y to a fresh z to get $\lambda z:A.[z \leftrightarrow y]m_1$, and then substitute m_2 in it to get $\lambda z:A.[m_2/x][z \leftrightarrow y]m_1$. The operation $[x \leftrightarrow y]m$ renames all free occurrences of y in m with x .

We define the substitution operation using a judgment $\mathcal{X} \vdash [m_1/x]m_2 = m_3$. The judgment means that if m_1 is a term with free variables in \mathcal{X} , and m_2 is a term with variables \mathcal{X}, x , then the substitution operation yields the term m_3 with variables under \mathcal{X} . Here \mathcal{X} denotes a set of variables.

$$\begin{array}{c}
\frac{}{\mathcal{X} \vdash [m/x]x = m} \text{(sub-var}_1\text{)} \qquad \frac{x \neq y}{\mathcal{X} \vdash [m/x]y = y} \text{(sub-var}_2\text{)} \\
\\
\frac{\mathcal{X}, z \vdash [m/x][z \leftrightarrow y]m' = m'' \quad z \# \mathcal{X}, x}{\mathcal{X} \vdash [m/x]\lambda y:A.m' = \lambda z:A.m''} \text{(sub-}\lambda\text{)} \\
\\
\frac{\mathcal{X} \vdash [m/x]m_1 = m'_1 \quad \mathcal{X} \vdash [m/x]m_2 = m'_2}{\mathcal{X} \vdash [m/x]\mathbf{apply}(m_1, m_2) = \mathbf{apply}(m'_1, m'_2)} \text{(sub-apply)} \\
\\
\frac{\mathcal{X} \vdash [m/x]m_1 = m'_1 \quad \mathcal{X} \vdash [m/x]m_2 = m'_2}{\mathcal{X} \vdash [m/x]\langle m_1, m_2 \rangle = \langle m'_1, m'_2 \rangle} \text{(sub-}\langle \rangle\text{)} \\
\\
\frac{\mathcal{X} \vdash [m/x]m' = m''}{\mathcal{X} \vdash [m/x]\mathbf{fst} m' = \mathbf{fst} m''} \text{(sub-fst)} \qquad \frac{\mathcal{X} \vdash [m/x]m' = m''}{\mathcal{X} \vdash [m/x]\mathbf{snd} m' = \mathbf{snd} m''} \text{(sub-snd)} \\
\\
\frac{\mathcal{X} \vdash [m/x]m' = m''}{\mathcal{X} \vdash [m/x]\mathbf{inl}_{B,C}m' = \mathbf{inl}_{B,C}m''} \text{(sub-inl)} \\
\\
\frac{\mathcal{X} \vdash [m/x]m' = m''}{\mathcal{X} \vdash [m/x]\mathbf{inr}_{B,C}m' = \mathbf{inr}_{B,C}m''} \text{(sub-inr)} \\
\\
\frac{\mathcal{X} \vdash [m/x]m_1 = m'_1 \quad \mathcal{X}, y' \vdash [m/x][y' \leftrightarrow y]m_2 = m'_2 \quad \mathcal{X}, z' \vdash [m/x][z' \leftrightarrow z]m_3 = m'_3 \quad y' \# \mathcal{X}, x \quad z' \# \mathcal{X}, x}{\mathcal{X} \vdash [m/x]\mathbf{case} m_1\{\mathbf{inl} y \Rightarrow m_2 \mid \mathbf{inr} z \Rightarrow m_3\} = \mathbf{case} m'_1\{\mathbf{inl} y' \Rightarrow m'_2 \mid \mathbf{inr} z' \Rightarrow m'_3\}} \text{(sub-case)}
\end{array}$$

Substitution formalizes the idea of *plugging in* proof terms for hypotheses. The following principle defines the meaning of hypothetical judgments (Γ^- denotes the set of variables in Γ):

If $\Gamma, x:A \vdash m_2:B$ and $\Gamma \vdash m_1:A$, and $\Gamma^- \vdash [m_1/x]m_2 = m_3$ then $\Gamma \vdash m_3:B$.

Instead of making this principle an inference rule in our system, we prove it as a theorem about our system.

Task 4 (Substitution principle for proof terms) *Show that the following rule is admissible*

$$\frac{\Gamma, x:A \vdash m_1:B \quad \Gamma \vdash m_2:A \quad \Gamma^- \vdash [m_2/x]m_1 = m'_1}{\Gamma \vdash m'_1:B} \text{(subst)}$$

Since we did not formally define the renaming operation $[x \leftrightarrow y]m$, you are free to use the following Lemma:

Lemma 2.1 *If $\Gamma, x:A \vdash m:B$ and $y \# \Gamma$, then $\Gamma, y:A \vdash [y \leftrightarrow x]m:B$.*

(Hint) *To prove Task 4, observe that substitution is defined by induction on the term being substituted into. Proceed by induction on the derivation of $\Gamma \vdash [m_1/x]m_2 = m_3$. You can use the following lemma to reason backwards about typing of proof terms:*

Lemma 2.2 (Inversion) *If $\Gamma \vdash m:A$ then one of the following holds:*

- *If $m = x$, $\Gamma = \Gamma_1, x:A, \Gamma_2$*
- *If $m = \lambda x:A_1.m_1$, then $A = A_1 \supset A_2$, and $\Gamma, x:A_1 \vdash m_1:A_2$ is a strict sub-derivation.*
- *If $m = \mathbf{apply}(m_1, m_2)$, then $\Gamma \vdash m_1:A_2 \supset A$ and $\Gamma \vdash m_2:A_2$ are strict sub-derivations.*
- *If $m = \langle m_1, m_2 \rangle$, then $A = A_1 \wedge A_2$, and $\Gamma \vdash m_i:A_i$ for $i = 1, 2$ are strict sub-derivations.*
- *If $m = \mathbf{fst} \ m'$, then $\Gamma \vdash m':A \wedge A'$ is a strict sub-derivation.*
- *If $m = \mathbf{snd} \ m'$, then $\Gamma \vdash m':A' \wedge A$ is a strict sub-derivation.*
- *If $m = \mathbf{inl}_{A_1, A_2} m'$, then $A = A_1 \vee A_2$ and $\Gamma \vdash m':A_1$ is obtained as a strict sub-derivation.*
- *If $m = \mathbf{inr}_{A_1, A_2} m'$, then $A = A_1 \vee A_2$ and $\Gamma \vdash m':A_2$ is obtained as a strict sub-derivation.*
- *If $m = \mathbf{case} \ m' \{ \mathbf{inl} \ x \Rightarrow m_1 \mid \mathbf{inr} \ y \Rightarrow m_2 \}$, then the following occur as strict sub-derivations:*
 1. $\Gamma \vdash m':A_1 \vee A_2$,
 2. $\Gamma, x:A_1 \vdash m_1:A$,
 3. $\Gamma, y:A_2 \vdash m_2:A$,

It can be shown that all terms m s.t. $\mathcal{X} \vdash [m_1/x]m_2 = m$ differ from each other only in the choice of bound variable names, i.e. they are α -equivalent. From now on we shall use the term $[m_1/x]m_2$ to denote any arbitrary term in this set.

3 Reduction of proof terms

The central organizing principle of natural deduction is that intro-rules and elim-rules should “invert” each other. In this question, we shall only consider the inversion of intro-rules by elim-rules. This gives rise to a notion of conversion of proof terms. We define a binary relation \mapsto_β (β -conversion) between proof terms that witnesses this relationship:

$$\begin{array}{ll}
\mathbf{apply}(\lambda x:A.m_1, m_2) & \mapsto_\beta [m_2/x]m_1 \quad (\supset_\beta) \\
\mathbf{fst} \langle m_1, m_2 \rangle & \mapsto_\beta m_1 \quad (\wedge_{\beta,1}) \\
\mathbf{snd} \langle m_1, m_2 \rangle & \mapsto_\beta m_2 \quad (\wedge_{\beta,2}) \\
\mathbf{case} \mathbf{inl}_{A,B} m \{ \mathbf{inl} \ x \Rightarrow m_1 \mid \mathbf{inr} \ y \Rightarrow m_2 \} & \mapsto_\beta [m/x]m_1 \quad (\vee_{\beta,1}) \\
\mathbf{case} \mathbf{inr}_{A,B} m \{ \mathbf{inl} \ x \Rightarrow m_1 \mid \mathbf{inr} \ y \Rightarrow m_2 \} & \mapsto_\beta [m/y]m_2 \quad (\vee_{\beta,2})
\end{array}$$

We have used the notation $[m/x]m'$ to denote the set of terms m'' such that $\Gamma \vdash [m/x]m' = m''$ for some Γ, A .

On the left are proof terms that contain an introduction form (e.g. $\lambda x:A.m$) immediately followed by an elimination form (e.g. $\mathbf{apply}(m_1, m_2)$).

In the following, you can again assume Inversion Lemma 2.2 without proving it.

Task 5 (Local soundness) *In this question, we shall see why the relation \mapsto_β is justified. Each clause $m \mapsto_\beta n$ in the above definition simplifies the proof m to n either by removing a detour, or by using substitution. Your task in this question is to explain this simplification.*

To proceed, consider each clause $m \mapsto_\beta n$. Using Lemma 2.2, draw a derivation tree for $\Gamma \vdash m:A$. Then draw another proof tree \mathcal{E} that corresponds to n for the derivation $\Gamma \vdash n:A$, observing how the proof is simplified. As an illustration, we show the case for $\mathbf{apply}(\lambda x:A.m, m') \mapsto_\beta [m'/x]m$.

$$\frac{\frac{\Gamma, x:A \vdash m:A' \quad \Gamma \vdash m':A}{\Gamma \vdash \lambda x:A.m:A \supset A'} (\supset\text{-I}^+)}{\Gamma \vdash \mathbf{apply}(\lambda x:A.m, m'):A'} (\supset\text{-E}^+) \mapsto_\beta \frac{\Gamma, x:A \vdash m:A' \quad \Gamma \vdash m':A}{\Gamma \vdash [m'/x]m:A'} (\text{subst})$$

3.1 Alternative rules for conjunction

We will now consider an alternative formulation of rules for conjunction formation and elimination.

$$\frac{\Gamma \vdash m_1:A_1 \quad \Gamma \vdash m_2:A_2}{\Gamma \vdash \langle m_1, m_2 \rangle:A_1 \wedge A_2} (\wedge\text{-I}')$$

$$\frac{\Gamma \vdash m:A_1 \wedge A_2 \quad \Gamma, x:A_1, y:A_2 \vdash m':B \quad x\#\Gamma \quad y\#\Gamma}{\Gamma \vdash \mathbf{let} \ m = \langle x, y \rangle \ \mathbf{in} \ m':B} (\wedge\text{-E}')$$

While the introduction rule is the same, the new formulation uses a single elimination rule instead of two used in the earlier formulation.

Task 6 (β -conversion for the new formulation) We will now work out a β -conversion rule for pairs in the new formulation. To begin with, observe that Rule (\wedge -I') immediately followed by Rule (\wedge -E') corresponds to the proof term $\text{let } \langle m_1, m_2 \rangle = \langle x, y \rangle \text{ in } m$. Your task is to fill in the following blank:

$$\text{let } \langle m_1, m_2 \rangle = \langle x, y \rangle \text{ in } m \mapsto_{\beta} \dots\dots\dots$$

(Hint) Think how one would simplify the proof tree corresponding to the above term.

3.2 Reduction

β -conversion induces a notion of reduction of terms. The reduction relation \rightsquigarrow is obtained by closing β -conversion under transitivity and congruence. For this question, we consider a fragment of the logic consisting only of implication (we correspondingly consider only Rule (\supset_{β})). The definition of reduction is shown in Figure 1.

β -conversion

$$\frac{}{\text{apply}(\lambda x:A.m, n) \mapsto_{\beta} [n/x]m} (\supset_{\beta})$$

Single step

$$\frac{m \mapsto_{\beta} m'}{m \rightsquigarrow m'} (\text{step})$$

Transitivity

$$\frac{m_1 \rightsquigarrow m_2 \quad m_2 \rightsquigarrow m_3}{m_1 \rightsquigarrow m_3} (\rightsquigarrow\text{-trans})$$

Congruence

$$\frac{m \rightsquigarrow n}{\lambda x:A.m \rightsquigarrow \lambda x:A.n} (\lambda \rightsquigarrow) \quad \frac{m_1 \rightsquigarrow m'_1}{\text{apply}(m_1, m_2) \rightsquigarrow \text{apply}(m'_1, m_2)} (\text{apply } \rightsquigarrow_L)$$

$$\frac{m_2 \rightsquigarrow m'_2}{\text{apply}(m_1, m_2) \rightsquigarrow \text{apply}(m_1, m'_2)} (\text{apply } \rightsquigarrow_R)$$

Figure 1: Reduction of proof terms

Task 7 (Subject reduction) Prove that if $\Gamma \vdash m:A$ and $m \rightsquigarrow m'$, then $\Gamma \vdash m':A$. You can use Lemma 2.2 without proving.

(Hint) Proceed by induction on the derivation of $m \rightsquigarrow m'$. Notice that in order to get through the Rule (step), you will need to prove that an analogous property holds for the relation \mapsto_β , i.e.

If $\Gamma \vdash m:A$ and $m \mapsto_\beta m'$ then $\Gamma \vdash m':A$.

You may want to prove this separately as a lemma. To prove this, proceed by induction on the derivation of $m \mapsto_\beta m'$. There is a single rule (\supset_β) that defines \mapsto_β in this question.